System Specification

Hypergolic Maintenance Facility (HMF) IPT, Thor DP1

Checkout and Launch Control System (CLCS)

84K00302-016

Hypergolic Maintenance Facility (HMF) Integrated Product Team (IPT)

Assessment

March 12, 1998

Version 2.0

HMF IPT
Table of Contents

1. Introduction 1
1.1 HMF IPT Overview 1
1.2 HMF IPT Concept 1
1.2.1 CLCS Real-Time Control Application Software Architecture Concept 1
1.2.2 HMF Set Architecture 4
1.2.3 HMF End Item Overview 5
1.2.4 HMF RTC Application Software Concept 5
1.3 HMF IPT Specification 7
1.3.1 Statement of Work 7
1.3.2 Requirements 7
1.4 HMF IPT Hardware Diagram 7
1.5 HMF IPT Deliverables 7
1.6 HMF IPT Assessment Summary 7
1.6.1 Labor Assessments 7
1.6.2 Hardware Costs 8
1.6.3 HMF IPT Procurement 8
1.7 HMF IPT Schedule & Dependencies 8
1.7.1 Schedule 8
1.7.2 Dependencies 8
1.8 HMF IPT Simulation Requirements 8
1.9 HMF IPT Integration and System Test 9
1.10 HMF IPT Training Requirements 9
1.10.1 Training Needed 9
1.10.2 Training to be provided 9
1.11 HMF IPT Facilities Requirements 9
1.12 Travel Requirements 9
1.13 HMF IPT Action Items/Resolution 9
2. CSCI Assessments 9
2.1 HMF CSCI Assessment 9
3. HWCI Assessments 10
4. COTS Products Dependencies 10
4.1 SW Products Dependency List 10
4.2 HW Products Dependency List 11

Assessment Team

Name	CI Represented	E-Mail Address	Phone
Frank Norris	HMF	Frank.Norris-1 @ksc.nasa.gov	1-4955
Rob Fain	HMF	rfain @clcsmail.ksc.nasa.gov	1-7563
Rich Ikerd	Application Product Team	Richard.Ikerd-1 @ksc.nasa.gov	1-7541
Robert Pierce	Application Product Team	Robert.G.Pierce @usago.ksc.nasa.gov	1-7505
Ken Hale	Application Product Team	Kenneth.Hale-1 @ksc.nasa.gov	1-2222

1. Introduction

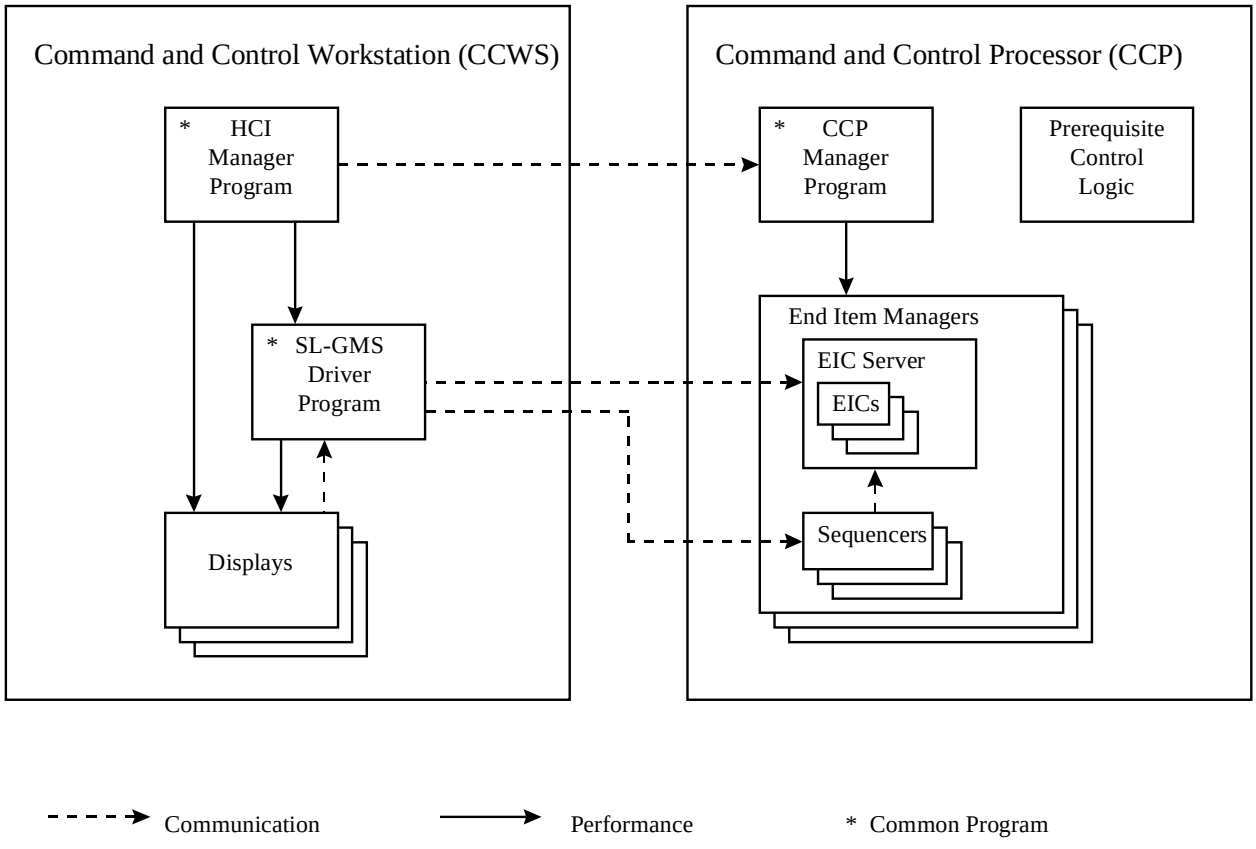
1.1 HMF IPT Overview

This Integrated Product Team (IPT) is responsible for the definition, design, and development of the Hypergolic Maintenance Facility (HMF) Real-Time Control Application Software. This includes software to support the check-out and maintenance testing of the Forward Reaction Control System (FRCS) and Aft Propulsion System (APS). The APS consists of the Left Orbital Maneuvering Subsystem (LOMS) and Right Orbital Maneuvering Subsystem (ROMS).

1.2 HMF IPT Concept

1.2.1 CLCS Real-Time Control Application Software Architecture Concept

A Real-Time Control (RTC) Application Software CSCI is distributed between a Command and Control Workstation (CCWS) and a Command and Control Processor (CCP). This distribution allows for processor loading and functionality to be controlled to enhance performance and maintainability. The following diagram illustrates the conceptual model for CLCS RTC Application Software. It identifies each of the major elements of the architecture and shows which element is associated with which processor. This distribution of functionality is designed to allow separation of display information and control functions. Display Programs are not directly tied to the execution of the commanding sequences and can be initiated/terminated independent of such sequences. The use of Display Programs will not impact the performance of command sequences or vice-versa. Note: This architecture concept will be common across all RTC Application Software CSCIs. Reference the RTC Application Software Architecture Standard for more information.



CCWS RTC Application Software

1. ***HCI Manager Program*** - The HMP is a single point interface for selecting which Display Programs the user wants to perform and for providing a list of all active programs the user has started.
   A. Executes in a CCWS, performed by user selection of a User Class from the CCWS Navigation System (CNS) menu.
   B. Provides health status to Subsystem Integrity.
   C. Provides controls to access all display programs available in an OCR. The associated User Class's displays are available from a set of primary controls.
   D. Maintains an activity status of the displays the user has started (e.g., active, in exception).
   E. Provides controls for quick access of executing displays programs (e.g., de-icon, pop to front)
   F. Provides a summary status (and a pop-up detailed status) of all End Item Managers associated with the primary User Class(es).
   G. Is a common application to be used by all RTC Application Software CSCIs. Can be customized using a configuration script to indicate associated primary User Class(es). No code modification of the HMP is required for use by a CSCI.

2. ***SL-GMS Driver Program*** - This program is required if the Application Software CSCI utilizes the COTS tool SL-GMS for developing dynamic data Display Programs.
   A. Executes in a CCWS, performed by the HMP upon request to perform the first SL-GMS display program. A request for a second display program is passed to the SL-GMS Driver for processing.
   B. Handles all of the tool specific functions necessary to correctly execute SL-GMS dynamic data displays.
   C. Is a common application to be used by all Application Software CSCIs. Will require minimum customization to become an integral part of the CSCI.
   D. Provides a single interface for data distribution into the SL-GMS display programs and for command request issuance from the display programs.
   E. Handles user prompt communication between EIM and CCWS.

3. ***Display Programs*** - These are the primary user interface programs used for monitoring all vehicle and GSE data, issuing commands to hardware end-items and for controlling automated sequences.
   A. Executes in a CCWS, performed by the SL-GMS Driver program for those programs developed using SL-GMS or by the HMP for non-SL display programs.
   B. CSCI can have as many Display Programs as necessary to support the required visibility and functionality for check-out and monitoring of the vehicle/GSE system.
   C. Dynamic data displays will be developed using the COTS tool SL-GMS.
   D. Non-dynamic data displays can be developed using standard languages and windowing techniques.
   E. Communicates with EIMs via the SL-Driver Program

CCP RTC Application Software

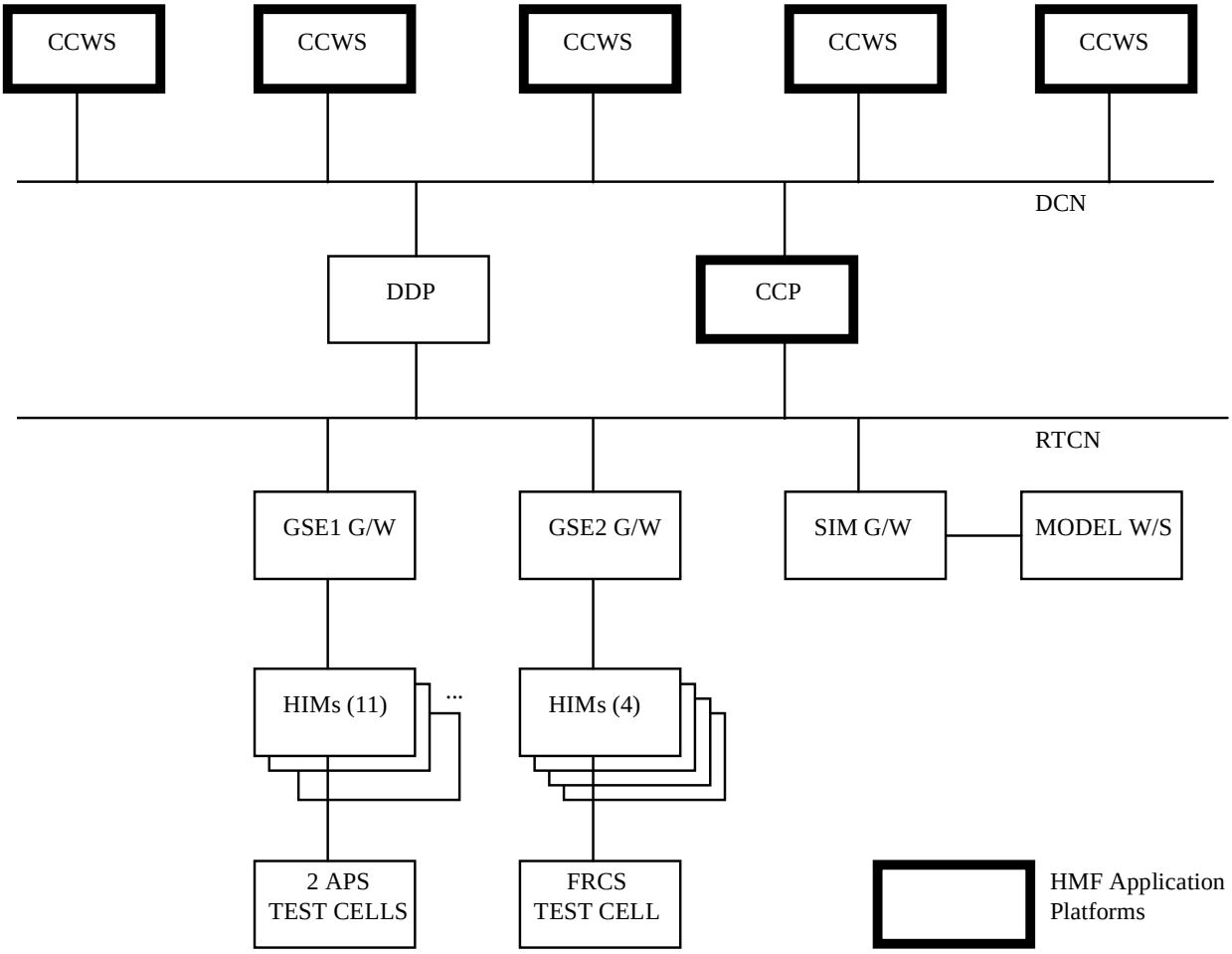
1. ***CCP Manager Program*** - The CMP is a single point interface for initializing all End Item Managers associated with an Application Software CSCI.
   A. Executes in a CCP, performed by either OPS CM at set initialization or by the Master Console at CCP initialization to support a test.
   B. There will be one CMP per CCP.
   C. Is merely a script or simple sequence execution program. Template will be common for all Application Software CSCIs, while contents will be customized.
   D. Initializes all associated EIMs using the Initiation and Termination Service.
   E. Upon successful completion of its task, the CMP terminates. All health/redundancy management of the EIMs then becomes the responsibility of Subsystem Integrity.

2. ***End Item Managers*** - If an Application Software CSCI is required to issue vehicle/GSE commands, it must have at least one EIM.
   - A. General Characteristics
     1. Executes in a CCP, performed by the CMP.
     2. Provides process health status to Subsystem Integrity.
     3. Handles all vehicle/GSE command requests from displays and other EIMs.
     4. Can be more than one EIM per Application Software CSCI.
     5. Can be hierarchical - control end items as individual components as well as subsystems (groups of components).
     6. Consists of one EIC Server and zero or more Sequencers.
   - B. End Item Component (EIC) Server
     1. Executes in a CCP, performed by the CMP.
     2. Is a single process that consists of multiple End Item Components.
   - C. End Item Components
     1. Is a single point interface for the control of a specific end item (e.g., valve).
     2. Handles all command requests for the end item component (e.g., open/close).
     3. Maintains the current state of the end item.
     4. Responds to exception notification (including Reactive Control Logic).
     5. Can be heirarchical - composed of other End Item Components.
     6. Multiple sequencers can reference the same EICs.
   - D. Sequencers
     1. Performs task-based operations on end item components.
     2. Provides an automated sequence mechanism encapsulated within an EIM.
     3. Can be implemented as a Finite State Machine.
     4. Controls individual components through the EICs.
     5. One or more sequencers can be in the same process.

3. ***Prerequisite Control Logic*** - PCL sequences are used to ensure an FD command is only issued if the specified conditions are satisfied.
   - A. PCL sequences are executed independent of the remainder of the Application Software.
   - B. All PCL sequences execute in a CCP, performed by the Command Management process.
   - C. If the conditions of the PCL are not satisfied the command request is rejected.

1.2.2 HMF Set Architecture

The CLCS HMF Set will consist of five Command and Control Workstations, a Data Distribution Processor, a Command and Control Processor, two GSE Gateways, and a Simulation Gateway. The GSE Gateways interface with Hardware Interface Modules (HIMs) which provide the connectivity to the hardware test cells (at the HMF, HIMs are used to interface with both GSE and flight hardware). These test cells contain the GSE hardware and the Orbiter Vehicle FRCS and OMS pods being serviced. Two test cells are used for OMS pod servicing (one for the left and one for the right OMS pod) and one test cell is used for FRCS servicing. The following figure illustrates the top-level architecture of the CLCS HMF Set equipment and shows that the HMF RTC Application Software platforms are the CCWS and CCP.

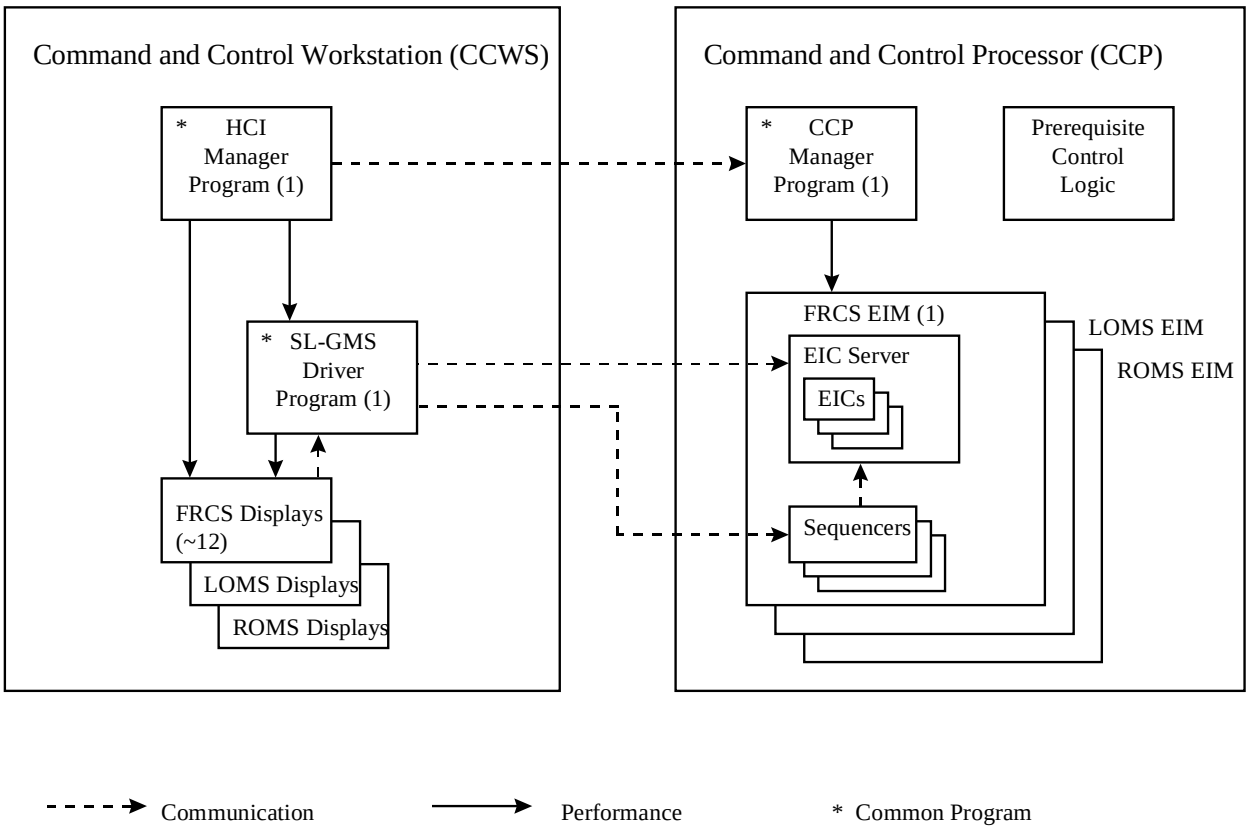


1.2.3 HMF End Item Overview

The HMF consists of GSE and vehicle hardware (FRCS, LOMS, and ROMS). Hardware across both GSE and vehicle can be generally separated into power-related subsystems and fluid or gas-related subsystems. Power subsystems include components such as power supplies, power buses, and power sub-buses. The fluid and gas subsystems include components such as heaters, manual valves, manifolds, GSE valves, vehicle valves, nodes, tanks, quick disconnects, regulators, saturators, screen test panels, scrubbers, and thrusters. The fluids and gases used at the HMF include gaseous helium, gaseous nitrogen, hypergolic fuel propellant (monomethyl hydrazine), and hypergolic oxidizer propellant (nitrogen tetroxide). There are approximately 4800 Function Designators associated with the HMF, 1000 of which are associated with the FRCS. There are an estimated 1800 end item components at the HMF, 500 of which are with the FRCS.

1.2.4 HMF RTC Application Software Concept

The HMF CSCI will follow the CLCS Real-Time Control Application Software architecture standard. The following figure provides an illustration of a possible architecture for the HMF RTC Applications. Detailed Functional Requirements have not been generated for the APS so a detailed design cannot yet be created. The HMF IPT is currently analyzing the HMF Pathfinder efforts so that a consolidated EIM design concept can be developed.



CCWS RTC Application Software

1. ***HCI Manager Program*** - One HMP will be developed for the HMF. This same program can be used by other CSCIs by customizing a configuration script.
2. ***SL-GMS Driver Program*** - One SL-GMS Driver program has been developed for the HMF. This program will require minimum customization to be utilized by other CSCIs.
3. ***Display Programs*** - Three kinds of displays are currently being planned for the HMF: tabular, schematic, and sequencer control. For the FRCS, it is estimated that approximately 2 tabular, 10 schematic, and 14

sequencer control displays will be required. Additional displays of all three kinds will be required for the LOMS and ROMS. The following is a list of the possible tabular and schematic displays for the FRCS (there will be a display for each sequencer as well). Reference the HMF FRCS Functional Requirements Document for more details.

Schematic Displays:
1. Oxidizer Helium System
2. Fuel Helium System
3. Oxidizer Propellant System
4. Fuel Propellant System
5. Thruster System
6. Oxidizer Scrubber
7. Fuel Scrubber
8. Oxidizer Screen Test
9. Fuel Screen Test
10. Power

Tabular Displays:
1. Heaters
2. Instrumentation

CCP RTC Application Software

1. ***CCP Manager Program*** - One CMP will be developed for the HMF. This program will require some customization to be utilized by other CSCIs.
2. ***End Item Managers*** - It is currently estimated that there will be one EIM each for the FRCS, LOMS, and ROMS at the HMF. Each EIM will consist of an EIC Server, End Item Components and sequencers. Actual numbers of components and sequencers will depend on the EIM design currently being developed by the IPT. The following estimates are based on the HMF FRCS Functional Requirements Document. Reference this document for more details.
   A. FRCS Components - approximately 30 types, 500 individual components
   B. FRCS Sequencers - approximately 14 including the following:
      1. Power Up/Down Power Supply
      2. Power Up/Down Sub Bus
      3. Scrubber Auto On/Off
      4. Primary Thruster Valve Signatures
      5. Vernier Thruster Valve Signatures
      6. Helium Tank Pressurization
      7. Helium Tank Vent
      8. Propellant Tank Pressurization
      9. Propellant Tank Vent
      10. Back Up Regulators and Open Helium Isolation Valves
      11. Propellant Tank Pulse Purges
      12. Manifold Pulse Puges
      13. Generis GSE Purge
      14. Regulator Flow

1. ***Prerequisite Control Logic*** - It is planned for the HMF to have more PCL protection than exists with the current CCMS HMF software. The amounts of PCL are TBD.

1.3 HMF IPT Specification

1.3.1 Statement of Work

The completion of this IPT's software development will be in the post-Thor timeframe. Software will be developed during Thor, but this software will not go through formal validation until after Thor. The following Statement of Work applies only to those items which will be worked for Thor.

- Continue development of FRCS application software (development was initiated as part of the HMF Pathfinder Thread during the Redstone period):
    - Finalize FRCS Functional Requirements Document
    - Generate the FRCS sections of the HMF Software Design Specification
    - Develop the HCI and CCP Manager programs
    - Complete FRCS Displays development
    - Continue design and development of FRCS End Item Managers (EIMs)
    - Integrate EIMs and Displays with Application Services for inter-process communications
- Begin development of APS Functional Requirements Document

1.3.2 Requirements

Detailed Application Software requirements for the FRCS are documented in the HMF Functional Requirements Document. Part of the Thor work will be beginning the development of requirements for the APS which will also be incorporated into the HMF Functional Requirements Document.

1.4 HMF IPT Hardware Diagram

Not Applicable.

1.5 HMF IPT Deliverables

Deliverable	R&D Document	Code	API Manual	Users Guide
FRCS CSC	Y	Y - see Note	N/A	N

Note: Software will be developed during Thor, but this software will not go through formal validation until after Thor.

1.6 HMF IPT Assessment Summary

This section contains the summary of the costs and labor involved in implementing the capability. It is broken into three sections. The first is a summary of the individual CSCI labor assessments. The second is a summary of hardware costs. The third is a summary of procurement activities needed.

1.6.1 Labor Assessments

The total Labor Costs required to provide this capability are summarized in the following table:

No.	CSCI/HWCI Name	Thor LM	Changes covered in
1	HMF CSCI	32	
	TOTAL	32	

1.6.2 Hardware Costs

None.

1.6.3 HMF IPT Procurement

None.

1.7 HMF IPT Schedule & Dependencies

1.7.1 Schedule

Task Name	Start	Finish
Thor Key Dates		
Thor Assessment Kickoff	9/29/97	11/7/97
Concept Panel Internal Review		11/7/97
Concept Panel		TBD
FRCS Requirements Review		TBD
FRCS Design Review		TBD
Thor Development		
FRCS Functional Requirements Document Finalized	11/3/97	12/19/97
FRCS Display Development	4/1/97	12/19/97
FRCS End Item Manager Development	12/1/97	3/13/98
FRCS EIM Interim Release (non-validated)		2/16/98
HCI and CCP Manager Program Development	7/1/97	1/15/98
APS Functional Requirements Document Definition	11/15/97	4/15/98
Thor Development Complete		3/27/98
Post-Thor Development		
APS Display Development	3/2/98	7/15/98
APS End Item Manager Development	3/2/98	10/30/98
APS EIM Interim Release (non-validated)		9/15/98
Validation	11/2/98	2/1/99

1.7.2 Dependencies

No.	Dependency Area	Dependency	Need Date
1	Requirements	Requirements Capture Tool	11/3/97
2	App. Services	Application Services API - Interfaces Defined	12/5/97
3	App. Services	Application Services API - Prelim. API Available	1/5/98
4	App. Services	Application Services API - Thor API Available	2/15/98
5	IPC Services	IPC Services - CORBA Method Confirmed	12/1/97
6	IPC Services	IPC Services - COTS Tool Selected	12/1/97
7	IPC Services	IPC Support Complete	2/2/98
8	Dev. Environment	HMF SGOS Math Model Connectivity	2/1/98
9	Dev. Environment	SDE/IDE Loaded with Thor Capabilities	2/15/98
10	Test Build	HMF TCID Built with Thor Baseline	2/1/98
11	Regression Test	Test Methodology Defined	3/30/98

1.8 HMF IPT Simulation Requirements

- The capability to test Application Software with the HMF Model using the Application Desktop Debug Environment, Simulation Gateway, CDS Math Model, or Rehosted Math Model is required for software development and debug.

- The capability to test Application Software with the HMF Model via VSI is required for validation in an integrated environment.

1.9 HMF IPT Integration and System Test

Not Applicable.

1.10 HMF IPT Training Requirements

1.10.1 Training Needed

- Requirements capture tool
- Control Shell 6.0 Beta

1.10.2 Training to be provided

None.

1.11 HMF IPT Facilities Requirements

None.

1.12 Travel Requirements

None.

1.13 HMF IPT Action Items/Resolution

- Explore the use of G2 in the implementation of advanced data analysis/expert systems for the HMF.

2. CSCI Assessments

The HMF IPT consists of one CSCI, the HMF CSCI.

2.1 HMF CSCI Assessment

Work Required

- FRCS Functional Requirements Document
- FRCS sections of the HMF Software Design Specification
- HCI and CCP Manager programs
- FRCS Displays development
- FRCS End Item Managers (EIMs)
- APS Functional Requirements Document (preliminary)

CSCI Assessment

Product	Labor (LM)	% of CSCI
Displays	6.92	40%
Command/Control	9.32	40%
Sequencers	14.72	40%
PCL	.28	40%

Product	Labor (LM)	% of CSCI
Total	31.24	40%

Basis of estimate

Anticipated labor costs for developing the HMF RTC Application Software CSCI are based on the current CCMS GOAL lines of code (executable) coupled with a complexity factor and input into a simple estimation formula. These estimates are for development of the entire HMF CSCI.

The formula used for estimation is: (c * KLOC) ** 1.12

Pgm Type	***GOAL ISN***	***Coefficient***	***Labor Month***
Display	51000	0.25	17.3
Command/Control	22200	0.75	23.3
Sequencer	20000	1.25	36.8
PCL	960	0.75	0.7
Totals			**78.1 LM**

- Note : The Sequencer GOAL ISN is an estimate since the HMF CCMS software does not contain any automated sequences, but they are planned for the CLCS HMF CSCI.

Since the HMF IPT effort for the Thor time-frame will only be addressing the Forward RCS systems and developing the CSCI infrastructure, the Thor labor estimate is approximately two-fifths of the total effort or 31.24 Labor Months.

Documentation

Document Type	New/Update	Number of Pages
Functional Requirements Document	Update (FRCS) New (APS)	TBD
Software Design Specification	New	TBD

Assumptions

None.

Open Issues

None.

3. HWCI Assessments

None.

4. COTS Products Dependencies

4.1 SW Products Dependency List

- Requirements capture tool
- SL-GMS (licenses already procured)
- Control Shell 6.0 Beta Release (licenses already procured) required 1/19/98
- CORBA COTS product for inter-process communications (vendors under evaluation)

4.2 HW Products Dependency List

None.